USACERL Technical Report M-90/15
July 1990
Automated Pavement Evaluation

US Army Corps
of Engineers
Construction Engineering
Research Laboratory

AD-A224 176

AutoPAVER, a Software Package for Automated Pavement Evaluation

by
Mark D. Ginsberg
M. Y. Shahin
Jeanette A. Walther

This research developed a method that improves data collection and reduces data entry times for Pavement Condition Index (PCI) surveys for use with PAVER, a pavement maintenance management system. The method, AutoPAVER, is a microcomputer software package used to analyze pictures of pavement surfaces and to forward the resulting analysis to PAVER. The user works interactively with the system to identify and classify pavement distresses. Distress measurement and data entry are done by the computer.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	July 1990	Final

4. TITLE AND SUBTITLE

AutoPAVER, a Software Package for Automated Pavement Evaluation

5. FUNDING NUMBERS

PR AT41
TASK MB
WU DA9

6. AUTHOR(S)

Mark D. Ginsberg, M.Y. Shahin, and Jeanette A. Walther

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

U.S. Army Construction Engineering Research Laboratory (USACERL)
2902 Newmark Drive, PO Box 4005
Champaign, IL 61824-4005

8. PERFORMING ORGANIZATION REPORT NUMBER

TR M-90/15

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

U.S. Army Engineering and Housing Support Center
ATTN: CEHSC-FB-P
Fort Belvoir, VA 22060

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

Copies are available from the National Technical Information Service, 5285 Port Royal Road, Springfield, VA 22161

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

This research developed a method that improves data collection and reduces data entry times for Pavement Condition Index (PCI) surveys for use with PAVER, a pavement maintenance management system. The method, AutoPAVER, is a microcomputer software package used to analyze pictures of pavement surfaces and to forward the resulting analysis to PAVER. The user works interactively with the system to identify and classify pavement distresses. Distress measurement and data entry are done by the computer.

14. SUBJECT TERMS

AutoPAVER
PAVER
Pavement condition index

15. NUMBER OF PAGES

25

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	SAR

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std 239-18
298-102

FOREWORD

This project was conducted for the U.S. Army Engineering and Housing Support Center (USAEHSC), under project 4A162731AT41, Task MB, Work Unit DA9, "Automated Pavement Evaluation/Dynamic Modeling of Pavement Performance/Maintenance Costs." Mr. Ken Gregg, CEHSC-FB-P, was the Technical Monitor.

The work was performed by the Engineering and Materials (EM) Division, U. S. Army Construction Engineering Research Laboratory (USACERL). The USACERL Principal Investigator was Dr. M. Y. Shahin. Dr. Paul A. Howdyshell is Acting Chief of the EM Division. The USACERL Technical Editor was William J. Wolfe, Information Management Office.

COL Everett R. Thomas is Commander and Director of USACERL, and Dr. L.R. Shaffer is Technical Director.

CONTENTS

AUTOPAVER, A SOFTWARE PACKAGE FOR AUTOMATED PAVEMENT EVALUATION

1 INTRODUCTION

Background

PAVER[1] is a field-tested and validated pavement maintenance management system for use by military installations, U.S. Army Corps of Engineers (USACE) Districts, airports, and civilian municipalities. PAVER is designed to optimize the allocation of funds for pavement maintenance and repair. PAVER's key component is the Pavement Condition Index (PCI), an objective rating of pavement condition based on the quantity and severity of observed distress. The PCI provides a consistent measure of a pavement's structural integrity and operational condition.

Information about a pavement's condition is usually gathered by someone who must view the pavement, determine which distresses are present, measure the size of the affected area, and record all information for later use by PAVER. This process is the most time consuming part of using the PAVER system. In order to accumulate accurate data, a field crew must travel to the site and cordon off the area for painstaking measurements. Traffic must be routed around the inspection area. The crew must visually inspect the affected areas, classify the pavement damage, and, using a tape measure, gauge the exact dimensions of each distress. If the pavement is cracked, the crew categorizes the type of crack and measures the length and width of the cracked area. Other types of distresses demand different types of measurement. Holes, for instance, require measurements for subsequent computation of surface area. The process as a whole is not only slow and cumbersome, but physically dangerous to the crew, since it must work adjacent to the flow of traffic. A method is needed to improve the efficiency of data collection and to reduce data entry times for PCI surveys.

Objectives

The objectives of this work were: (1) to develop a microcomputer software package that analyzes pictures of pavement surfaces and forwards the resulting information to PAVER, (2) to describe how image processing can be implemented in pavement analysis, and (3) to describe the capabilities of AutoPAVER by outlining its user interface.

[1] M. Y. Shahin, K. A. Cation, and M. R. Broten, *Pavement Maintenance Management: the Micro Paver System*, Technical Report M-87/12/ADA187360 (U.S. Army Construction Engineering Research Laboratory [USACERL], July 1987); D. R. Uzarski and R. C. Soule, *The Practical Use of PAVER in Planning, Programming, and Developing Projects for Pavement Maintenance and Repair*, Technical Report M-86/04/ADA167312 (USACERL, March 1986); M. Y. Shahin and S. D. Kohn, *Overview of the "PAVER" Pavement Management System and Economic Analysis of Field Implementing the "PAVER" Pavement Management System*, Technical Manuscript M-310/ADA116311 (USACERL, January 1982).

Approach

AutoPAVER simplifies the workload of gathering and transmitting data to the PAVER program. Pavement sections are photographed and the photos are then fed into an image-processing system. The user works interactively with the software program by categorizing pavement damage, and then by identifying and outlining pavement distresses in the screen image. AutoPAVER uses sophisticated algorithms to measure the user-defined distresses and then performs the necessary data entry.

Mode of Technology Transfer

AutoPAVER will be made available to PAVER users through U.S. Army distribution centers.

2 IMAGE-PROCESSING TECHNOLOGY

Image processing is the science of modifying and analyzing pictures. The basic goals of image processing include enhancement or modification of an image to improve its appearance or highlight information, measurement of image elements, classification or matching of image elements, and recognition of items in an image.

AutoPAVER relies on image-processing techniques to help analyze and transform an image into useful PAVER information. Image processing is performed using step-by-step procedures called algorithms. These algorithms are often implemented using computers that are flexible and have relatively low processing and memory costs. The algorithms are expressed as, and become nearly synonymous with, programs for the computer. An algorithm can specify operations such as how to acquire the image. Special image-processing hardware often supplements the computer.

Terminology for Image Processing

Before discussing the details of image processing, it is important to provide a frame of reference with respect to the terminology used in this report. Because so many branches of science have used image processing independently, people have invented many different terms that describe the same ideas. This report describes applications of computer-assisted image processing using terminology from the computer science and electrical engineering disciplines.

Image-Processing Hardware

AutoPAVER requires electronic equipment suitable for acquiring, processing, and displaying monochrome images. The necessary hardware consists of a camera, a frame grabber (a digitizing circuit with a section of computer memory dedicated to holding an image), a computer that can access image memory, and a video monitor to display the contents of the memory (i.e., the picture being processed).

The digitizing circuit places a camera image into the image memory. This process involves digitizing a video frame by breaking it into an array of digital intensity values called pixels (short for "picture elements"). If the image is already represented as pixels, as in a computer graphic, image acquisition consists of simply moving the image from disk to the image memory.

The computer reads and writes information in image memory. The display device reads image memory and shows a representation of it. An entire video image stored in image memory is often called a "frame buffer" or "frame store." The most common frame size is derived from the National Television Standard for Color (NTSC) which results in a frame size of 512 by 480 pixels.

Most low cost computers work with data that is represented by eight bits (binary digits). For AutoPAVER, each pixel has an eight-bit byte, which yields 2^8 (256) different shades of gray. Zero represents black, 255 represents white, and all the shades of gray fall in the range between zero and 255.

The computer manipulates the pixels in the image memory. The display device converts processed pixels back into spatially organized image intensities. This display device is usually a digital-to-analog (D/A) converter that drives a monochrome or color television monitor.

The computer used to develop and run AutoPAVER is an 80386 personal computer, fitted with an EGA monitor. The second monitor, used for graphics viewing and enhancement, is a Sony PVM-1271Q. The frame grabber is Image Technology Inc.'s PC Vision Plus, and the camera used is a Sony X6-57 (CTD). It is worth noting that the CCD (Charge Coupled Device) feature suits a camera to this type of research, since it allows an image to be viewed on a computer monitor for a great length of time without the danger of "burning the image" permanently into the screen display. The pointing device used in this study is a Microsoft Mouse.[2]

Classifying Algorithms

Image-processing algorithms can be classified in several ways. If an algorithm changes a pixel's value, it is called a "point process." If the algorithm changes a pixel's value based on the values of that pixel and neighboring pixels, it is called an "area process." If the algorithm changes the position or arrangement of the pixels, it is called a "geometric process." Algorithms that change pixel values based on a comparison of two or more images are called "frame processes" (because individual video images are called "frames").

Image measurement is based on some broad assumptions about what items will appear in the picture. Classification of objects is relatively simple; recognition requires more knowledge (and assumptions) about what can appear in an image. For example, the number of pixels in an image of pavement within a certain range of values can simply be counted. If it is known that bleeding asphalt corresponds to these values, the image can be classified into bleeding and nonbleeding areas. If the machine is provided with further knowledge about the structure of bleeding asphalt, it might be able to recognize these areas in the image. Color may also be used to highlight the recognized areas.

Based on this classification framework and the AutoPAVER hardware, some example algorithms will be described. These examples are not necessarily exhaustive; they are intended only to facilitate an understanding of the image-processing options for this technology.

Point Processes

A point process algorithm scans through the image area using the pixel value at each point to compute a new value at that point.

Optical Negative

To produce the photographic negative of an image, the computer is programmed to take each pixel value f ($0 <= f <= 255$) and replace it with g ($g = 255 - f$).

Brightness

Point processes also can be used to enhance or modify pixel values. For example, adding 40 to each pixel value brightens the image and could improve the picture's appearance.

If the pixel value and its location are used, then a point process can be used to correct shading or smoothly change pixel values in an image area. Shading is an image artifact caused by slow spatial shifts

[2] Microsoft is a registered trademark of the Microsoft Corp, Redmond, WA.

in scene lighting or camera bias and sensitivity. A point process that computes the inverse of a shading function can eliminate or correct much of this shading.

By smoothly changing the pixel values in an area, the contrast of areas can be highlighted or adjusted. This function can produce results similar to the photographic darkroom techniques of burning and dodging (methods of adjusting local contrast).

Histogram Stretching

To calculate an "intensity histogram," the number of times a particular pixel value occurs in an image is counted. Using AutoPAVER hardware based on eight-bit pixels, 256 pixel values are possible. An image can be scanned and the number pixels with a given value counted. The result can be stored as one entry in a 256 place table. Such a table is called a "histogram."

The histogram is a type of image measurement. Because it examines a single pixel at a time, the histogram is the result of a point process that leaves the pixel's value unchanged.

Information provided by the histogram is useful for image enhancement and classification. If all pixel values are bunched in a small range (making the picture appear featureless), this information can be used to improve image contrast. Starting at intensity zero, the histogram can be searched for the first pixel value with more than a specified number of pixel counts. One might choose a value occurring in only a small area of the screen, say 30. Next, a similar search is performed starting at the highest pixel value. The region of the histogram between these two (low and high) values accounts for most of the pixels in the image. Then a point process is performed that sets pixel values below the low value to zero and above the high value to 255. This is sometimes called a "histogram clip." The pixels with intermediate values are adjusted so that they span the range of zero to 255.

Histogram stretching is a simple form of contrast enhancement. Note that the image has lost some information--the pixel values below the low value and above the high value have been set to constants. In general, image-processing operations lose some information in return for selecting or accentuating other information.

Notice that the histogram stretching algorithm uses three simpler algorithms: a histogram, a histogram clip, and a point process. Most algorithms are compounds of other algorithms. Therefore, the program writer must know which algorithms to apply and in what order to apply them to reach a processing goal.

Pseudocoloring

Pseudocoloring of a monochrome image is another example of a point process. In this case, the pixel value is the argument (input) for three different functions, and the output of these functions drives the red, green, and blue guns of a color monitor. This process allows a monochrome image to be colored by substituting any color for a particular shade of gray. Using eight-bit pixels, 256 colors can be displayed at once.

Area Processes

An area process uses information from neighboring pixels to modify pixel values or assert the existence of some property of the neighborhood. Area processes can generate a wide range of effects: spatial filtering (such as filtering out repeated elements), changing an image's structure, or "sharpening" the image's appearance by accentuating intensity changes. Other effects include: finding objects by matching images, measuring image properties, making assertions about object edges in the image, removing noise, and blurring or smoothing the image.

Convolution

Convolution is a classic image-processing algorithm commonly used for spatial filtering and finding image features. Since convolution requires many repeated calculations, and hence a great deal of computer memory, some subtle implementation issues must be considered.

The convolution operation replaces a pixel's value with the sum of that pixel's value plus those of its neighbors, each weighted (multiplied) by a factor. The weighting factors are called the "convolution kernel." The programmer labels the image points p (i, j) and the kernel (weighting) points k (x,y), where x and y range over values representing the relative placement of neighboring pixels.

Choose, as an example, a neighborhood consisting of a pixel plus its immediate eight neighbors (left, right, up, down, and diagonals). Label the center pixel as p (1,1). This center pixel is replaced by the linear sum of its neighbors times their respective weighting factors, k.

$$\begin{aligned} p(1,1) = {} & p(0,0) \times k(0,0) + p(1,0) \times k(1,0) + p(2,0) \times k(2,0) \\ & + p(0,1) \times k(0,1) + p(1,1) \times k(1,1) + p(2,1) \times k(2,1) \\ & + p(0,2) \times k(0,2) + p(1,2) \times k(1,2) + p(2,2) \times k(2,2) \end{aligned} \qquad \text{[Eq 1]}$$

To convolve an image, this operation is repeated at every pixel position in the image. This process is like sliding a kernel matrix over each row of pixels in the image matrix, and at each point, multiplying the kernel values with the image value "under" it, summing the result, and then replacing the pixel at the center of the kernel with that value. The equation then becomes:

$$\begin{aligned} p(i,j) = {} & p(i-1,j-1) \times k(0,0) + p(i,j-1) \times k(1,0) + p(i+1,j-1) \times k(2,0) \\ & + p(i-1,j) \times k(0,1) + p(i,j) \times k(1,1) + p(i+1,j) \times k(2,1) \\ & + p(i-1,j+1) \times k(0,2) + p(i,j+1) \times k(1,2) + p(i+1,j+1) \times k(2,2) \end{aligned} \qquad \text{[Eq 2]}$$

Implementation Issues

There are several interesting issues to be met here: Convolving an area of size X by Y with a kernel of size N by M requires X x Y x N x M multiplications and additions. Thus, a 512 by 480 image with a three by three kernel requires 2,211,840 multiplication/addition operations; this process can take a long time on a computer without fast multiplication hardware.

If the kernel is scanned over the image and replaces only the value under the center of the kernel at a given position, what happens to the edges of the image? For example, with the three by three kernel, a one-pixel border (box) is left around the image where pixels are not replaced. The convolution will always leave a border of "garbage" equal to half the kernel size around the image. This border of garbage may be ignored, set to zero, or have the nearest meaningful value copied into it.

The convolution on any pixel could result in a value larger than can be held in a pixel--possibly as large as the number of kernel elements times the number of bits in a pixel. In the generic system described previously, for example, that would be 3 x 3 x 256 = 2304, for which 12 bits per pixel would be needed--but there are only eight bits per pixel. Enough accuracy in the calculations must be kept to allow for this range. The convolution result can be scaled (e.g., each result divided by two) if it is to go back into the image memory.

In a related issue, kernel values, and therefore the convolution output, can be positive or negative. Negative intensity is mathematically useful, but since the program only recognizes values between zero and 255, negative intensities will not generate a screen display. The option to modify the convolution output desired to allow only positive, negative (negated to positive values), absolute values, or signed values as output. This also means that an additional sign bit must be kept in the calculations. Thus, for eight-bit pixels and a three by three kernel, 12 bits for the sum and an additional bit for the sign are needed.

Convolution itself is relatively simple, but the implementation issues complicate it. Unfortunately, this condition is true of most image-processing algorithms. For example, the issues of internal accuracy and what to do at the edge of the image appear in most other area processes. Any useful program must address these issues to implement and use the algorithms effectively.

Matched and Spatial Filters

Convolution applies to image processing by using matched or spatial filters. In a matched filter, the convolution kernel is essentially a small image of what is to be amplified or detected. For example, an edge is detected in an image by its sudden increase or decrease in image intensity. A convolution kernel for detecting a vertical edge itself looks somewhat like a vertical edge (Figure 1).

$$\begin{matrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{matrix}$$

Figure 1. A kernel matrix to detect a vertical edge.

Note the effect of the negative values: in a uniform image area, where all pixel values input to the convolution are the same, the convolution output will be zero (since the sum of any number times each of the 15 kernel elements is zero). The kernel has been padded with a vertical row of zeros to make it an odd size in both directions, so that it will detect both increases and decreases in intensity.

A similar kernel for amplifying horizontal edges would resemble a horizontal edge (Figure 2).

-1	-1	-1	-1	-1
0	0	0	0	0
1	1	1	1	1

Figure 2. A kernel matrix to detect a horizontal edge.

Larger kernels may include patterns that match similar patterns in the image (e.g., for the letter A). In this case, the kernel is a "template." Detection usually involves amplification of the desired feature followed by a yes/no question that asks, "Is the result above or below a certain threshold point?"

In a spatial filter, the convolution kernel reflects wave frequencies. Sound frequency is a count of the number of times per second a wave form repeats. Image frequency is the breakdown of an image into a series of sine and cosine waves. This breakdown can be done using a fast Fourier transform. The transform should be done both horizontally and vertically to reflect spatial frequencies in both directions.

A kernel can be built to select, and perhaps detect, a certain band of frequencies. Quickly changing image intensities are represented by high spatial frequencies, and slowly changing intensities, by lower spatial frequencies.

The following kernel could be used to select high spatial frequencies (Figure 3).

-1	-1	-1
-1	8	-1
-1	-1	-1

Figure 3. A kernel matrix to select high spatial frequencies.

This "Laplacian filter" approximates an unoriented second derivative operation (i.e., it measures a wavelength, the center frequency of which is no more than twice as large as the kernel). Because edges have high spatial frequencies (sudden intensity changes), this kernel selects edges of any direction and might be used as an "edge detector" for image analysis.

If the Laplacian kernel is slightly modified by making the center kernel element nine instead of eight, the filter would add the output of the Laplacian convolution to the original image (since a kernel with a one in the middle surrounded by zeros would yield the source image unchanged). This kernel selectively boosts high frequencies (edges) making the resulting image sharper and noisier. On the other hand, if a kernel were used that matches lower spatial frequencies, the image will be blurred.

The power of convolution lies in using information in a large area to make assertions about some property at individual image points. For example, the edge operators above sharpen the edges in an image by using the fact that physical edges are narrow and extend over some distance. The art of convolving lies in designing the right kernel, using a conjunction of experience and theory.

Nonlinear Area Processes

Convolution is relatively easy to implement, use, and analyze because it is a linear operation: it requires only sums of first degree products. Nonlinear operations, while a bit more difficult, are also useful and can be more powerful than a convolution. "Powerful" means either that they provide a better signal-to-noise ratio for detecting image elements or that they can detect features with less computation. Consider the following two examples of nonlinear area processes.

Sobel Filtering

A Sobel filter compares the result of two convolutions. The first convolution computes the degree to which an edge is oriented in the X direction, while the second computes the same for the Y direction. Simple trigonometry is used to estimate the strength and orientation of edge. The two kernels, X and Y, are shown in Figure 4.

$$X: \begin{matrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{matrix} \qquad Y: \begin{matrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{matrix}$$

Figure 4. Sobel filter kernel matrices.

Thus, the edge strength and orientation are represented by:

$$\text{Strength} = \text{sqrt}(X^2 + Y^2) \qquad \text{[Eq 3]}$$

$$\text{Orientation} = \arctan(Y/X)$$

This is a first derivative (oriented) edge finder, and the vector field it produces cannot be directly shown on a two dimensional image. The Sobel is a good edge detector and is frequently used as the first step in machine-vision algorithms. Because the Sobel algorithm uses a lot of computer memory, various approximations have been developed to implement it.

Median Filtering

A median filter replaces the pixel at the center of a neighborhood of pixels with the median of the neighborhood pixel values. The neighborhood values, including the center pixel, are sorted into ascending order, and the median (middle) value is used to replace the center pixel. The effect of a median filter is to remove spot and low level noise while retaining larger scale image features.

Pattern and Object Identification

Before it can classify image elements, the computer must first contain a definition of "image element." This definition can be elaborate, but a simple example will suffice: an element is a connected group of pixels. A "connected" pixel has the same value as a neighbor at 0, 90, 180, or 270 degrees.

To simplify the computation, the image can be "binarized"; that is, all pixel values above zero and below 255 can be converted. This is often done in machine vision since the process can control the lighting and objects in the image. The image is searched from top to bottom to locate elements with more than N (number of) connected pixels. The specified minimum of N pixels, eliminates the small spots of noise introduced by thresholding. Each element is labeled by changing its pixel value to a predetermined number. The number of pixels in an element can also be recorded for further classification.

Geometric Processes

Geometric processes change the spatial arrangement of pixels. They are often used to correct for distortions caused by camera optics or viewpoint. They also can enlarge a particular image area. Typical geometric processes rotate, stretch, and translate the image position. Other geometric processes can warp the image. Geometric algorithms can be expressed by a set of equations (a matrix) that maps a pixel at location x,y into a new address x',y'. For example, to rotate a square area clockwise by 90 degrees, the pixels are mapped by the following equation.

$$x' = (512 - 1) - y; \quad y' = x \qquad \text{[Eq 4]}$$

Because digital pixels are oriented in a strict checkerboard pattern, most geometric transforms are found to have gaps between the output pixels. If source pixels are placed in the destination area according to the transformation equations, the exact placement in the output image is rarely an integer. (This is one problem with digitized images that never occurs in continuous tone images such as photographic prints.)

Frame Processes

Algorithms that use more than one image are sometimes called "frame processes." A simple example is to subtract one image from another. The resulting differences can be used to compare the two images (e.g., to look for missing parts on a machine or circuit board, or to look at the same piece of pavement photographed several years apart to see the progression of damage). Frame processes can also be used to improve image quality and to detect motion.

If a television camera views a still object, a predefined number (N) of successive image frames can be summed to reduce noise introduced by the camera. This process requires a frame memory with enough bits per pixel to accommodate the sum. Dividing the sum by N produces an averaged image. If the noise is not correlated from frame to frame, the improvement in signal to noise will be of the order sqrt(N). A typical low cost video camera has about three bits of noise, so that averaging with an N of eight or 16 will noticeably improve the image.

3 AUTOPAVER FEATURES

AutoPAVER Menu Options

AutoPAVER includes a menu driven interface to guide the user through its various functions. Upon invoking AutoPAVER, the user sees a Main Menu bar across the top of the screen containing the eight Main Menu commands: PICTURE, LUTS, FILTERS, CONVOL, VIDEO, AUTOPAVER, MACROS, and CONFIG. By placing the mouse pointer on any one of the main menu commands and depressing and holding the mouse button, the user calls a pop up window that extends downward from the selected main menu entry, and that lists the submenu elements for that main menu command. By dragging the mouse pointer to the appropriate command and releasing the mouse button, the user selects relevant AutoPAVER operations. Table 1 shows each Main Menu command with its corresponding submenu.

Table 1

Main and Submenu Commands

PICTURE	LUTS	FILTERS	CONVOL	VIDEO	AUTOPAVER	MACROS	CONFIG
Quadrant	P Pseudocolor	High Pass	Sharpen	Offset/Gain	AP Load	Load	Save Global
Load	M Pseudocolor	Low Pass	Average	Vertical Wedge	AP Save	Save	Save Local
Save	Negative	Band Pass	Horizontal	Horizontal Wedge	AP Save As	Save As	Read Global
Save As	Hist Stretch	Band Cut	Laplacian	Upside down	Send	Watch Me	Read Local
Print	Hist Plot	Binarize	Lap4	Grab	Send To	Play Back	Read Path
Quit	Brighten	Restore	Median	Snap	Change Pvmnt		Picture Path
About AutoPAVER	Dim		Edge		Clear Sheet		AutoPAVER Path
	Unify		Sobel		Dimensions		PAVER Path
	Linearize				PAVER Tools		Macro path
					Drawing Tools		

The following section lists the AutoPAVER menu options with a brief description of each.

PICTURE

About AutoPAVER - Lists program's developers and gives point of contact.

Quadrant - Menu to choose which quadrant that "Unify" and CONVOL will modify.

Load - Menu to load picture from disk.

Save - Saves picture to disk in picture file last used.

Save As - Saves picture to disk under a new name.

Print - Sends picture to printer. Currently supports Epson-FX, which is a two color printer. Consequently it is best to use Binarize to adjust how the picture will look and then Unify to change frame store before printing.

Quit - Prompts to leave program.

LUTS (Changes Look Up Tables only)

P Pseudocolor - One method of pseudocoloring. Use Horizontal Wedge or Vertical Wedge to explore how this works.

M Pseudocolor - A second method of pseudocoloring. Use Horizontal Wedge or Vertical Wedge to explore how this works.

Negative - Produces an optical negative.

Hist Stretch - Histogram Stretch (explained above in section 2) causes overall picture to have better contrast.

Hist Plot - Posts up histogram of current quadrant of the screen in the current quadrant of the screen. This is arranged from zero (black) on the left to 255 (white) on the right.

Brighten - Increases all gray values.

Dim - Decreases all gray values.

Unify - Copies LUTS values onto frame store (includes filter effects).

Linearize - Clears LUTS tables to normal black and white.

FILTERS (Changes look up tables only)

High Pass - Leaves whiter values (above adjustable cutoff) untouched and maps darker values onto chosen replacement color.

Low Pass - Leaves darker values (below adjustable cutoff) untouched and maps lighter values onto chosen replacement color.

Band Pass - Leaves middle gray levels (between adjustable cutoffs) untouched and maps higher and lower values onto replacement color.

Band Cut - Leaves extreme gray levels (outside adjustable cutoffs) untouched and maps middle levels onto replacement color.

Binarize - Forces all colors (either side of adjustable cutoff) to be mapped to black and white.

Restore - Removes filter effects.

CONVOL (Convolutions)

Sharpen - Causes image to appear sharper by local contrast stretching. (User selects degree of stretching.)

Average - Causes image to appear fuzzier (or less noisy) by local contrast averaging.

Vertical - Vertical edge detector. Leaves bright areas near vertical edges and darkens other areas.

Horizontal - Horizontal edge detector. Leaves bright areas near horizontal edges and darkens other areas.

Laplacian - Edge detector approximates the magnitude of the second derivative.

Lap4 - Edge detector sensitive to horizontal and vertical edges. An alternate form of Laplacian.

Median - Takes median of pixel neighborhood (user selectable).

Edge - Edge detector approximates result from Sobel kernel (below) only faster

Sobel - Commonly used edge detector. Leaves thick indication lines thereby allowing easier segmentation of the image.

VIDEO

Offset/Gain - Allows user adjustment of offset and gain of the camera. (Offset and gain are analogous to the brightness and contrast knobs of a television set.)

Vertical wedge - Standard test pattern.

Horizontal wedge - Standard test pattern

Upside down - Turns image upside down.

Grab - Continuous view from camera.

Snap - Snaps a picture with the camera.

AUTOPAVER

AP Load - Menu to load AutoPAVER analysis file.

AP Save - Saves AutoPAVER analysis file under analysis file name last used.

AP Save As - Prompts for name of file and store AutoPAVER analysis.

Send - Sends PAVER data to filename last used.

Send To - Selects new filename and send PAVER data to that file.

Change Pvmnt - Toggles pavement type (asphalt/concrete).

Clear Sheet - Clears all analysis previously drawn by user.

Dimensions - Sets the dimensions of the image in the video screen.

PAVER Tools - Posts the PAVER distress types for current pavement type.

Drawing Tools - Posts up the various drawing tools.

MACROS (For often repeated operations.)

Load - Menu to select macro file.

Save - Saves current macro set to macro file last used.

Save As - Prompts user for macro file name and save current macros there.

Watch Me - Starts macro recording.

Play Back - Repeats a macro.

CONFIG

Save Global - Saves the current configuration to the global *configuration file* (i.e., the configuration used when AutoPAVER is executed from any drive or directory other than those with a local configuration).

Save Local - Saves the current configuration to the local configuration file (i.e., the configuration used when AutoPAVER is executed from that particular drive/directory).

Read Global - Loads the global configuration file.

Read Local - Loads the local configuration file.

Picture Path - Selects drive and path where picture files are kept.

AutoPAVER Path - Selects drive and directory where AutoPAVER analysis files are kept.

PAVER Path - Selects drive and path where PAVER data files kept.

Macro Path - Selects drive and path where macro files are kept.

Using AutoPAVER

The AutoPAVER system has the appearance and simplicity of a coloring book. The system requires two video displays, one for the AutoPAVER menu interface, and the other for the graphic display of pavement. On the first screen, the user selects commands and AutoPAVER operations that will affect the video display on the second screen. When the user slides the mouse pointer past the edge of the menu screen, the pointer reappears on the graphics screen. In this way, the user travels between the two screens, using the mouse pointer in the program screen to select commands, and then in the graphic screen, to outline distressed areas of the displayed section of pavement. The user views a photograph of a pavement section on the graphics monitor, indicates to the system what type of pavement distress appears in the picture, and marks the distress (on the video screen) with a pointing device (i.e., a mouse). The system performs all numerical calculations and automatically forwards this information to PAVER. The user makes no numerical measurements or calculations, and needs no manual numerical record keeping.

Using the CONFIG submenu, the user must enter the following information into AutoPAVER in order for the system to be properly configured:

1. The base memory address of the frame grabber
2. The base I/O address of the frame grabber
3. The picture drive/directory
4. The AutoPAVER analysis drive/directory
5. The PAVER data drive/directory
6. The Macro file drive/directory.

One subtlety in using AutoPAVER is that sometimes the data which makes up the frame buffer (how the computer sees the picture) is different than that which is shown on the data analysis screen. Whenever this happens, there will be a warning indicator on the second line of the computer screen.

Many times this is a desirable situation. It helps user to "try out" certain ways of displaying the data without changing the frame buffer. To delete the warning indicator, simply use the mouse pointer to click on the warning indicator icon, and a computer will prompt to change the frame buffer from the way the screen looks to a simple display of the frame buffer as the computer sees it (e.g., "Change Frame Buffer?"). Simply choose which display you want and the computer will take care of the rest.

4 CONCLUSIONS

A microcomputer software package, AutoPAVER, was developed that analyzes pictures of pavement surfaces and forwards the resulting information to the PAVER pavement management system. AutoPAVER accomplishes pavement analysis through the use of image-processing techniques and algorithms. The AutoPAVER system uses a commercially available computer equipment and is accessed through a menu-driven software interface.

The AutoPAVER system saves both time and labor since it relieves the user of dangerous traffic-blocking chores while examining pavement, and because it performs all numerical tallying and record keeping unassisted. Development of various methods for automatic detection of pavement distresses is under way and will be added to AutoPAVER as each becomes available.

AutoPAVER is beneficial as an intermediate step between a totally manual field inspection method and a totally automated image-processing system. The system offers three principal advantages:

1. Data entry time is eliminated and the time required to measure distresses is reduced.

2. There is no need to control or block traffic for this procedure since available commercial equipment can take pavement photographs at the speed of traffic.

3. A record of the pavement condition is kept on either 35 mm film or video tape. This will be useful for verifying data and also for future research.

USACERL DISTRIBUTION

Chief of Engineers
ATTN: CEHEC-IM-LH (2)
ATTN: CEHEC-IM-LP (2)
ATTN: CECC-P
ATTN: CECW
ATTN: CECW-O
ATTN: CECW-P
ATTN: CECW-RR
ATTN: CEMP
ATTN: CEMP-C
ATTN: CEMP-E
ATTN: CERD
ATTN: CERD-L
ATTN: CERD-C
ATTN: CERD-M
ATTN: CERM
ATTN: DAEN-ZCE
ATTN: DAEN-ZCI
ATTN: DAEN-ZCM
ATTN: DAEN-ZCZ

CEHSC
ATTN: CEHSC-ZC 22060
ATTN: DET III 79906
ATTN: CEHSC-F 22060
ATTN: CEHSC-FB-P 22060
ATTN: CEHSC-TT-F 22060

US Army Engineer Districts
ATTN: Library (41)

US Army Engineer Divisions
ATTN: Library (14)

ROK/US Combined Forces Command 96301
ATTN: EUSA-HHC-CFC/Engr

US Military Academy 10996
ATTN: Facilities Engineer
ATTN: Dept of Geography & Computer Sciences
ATTN: MAEN-A

AMC - Dir., Inst., & Svcs.
ATTN: DEH (23)

DLA ATTN: DLA-WI 22304

DNA ATTN: NADS 20305

FORSCOM (28)
FORSCOM Engineer, ATTN: Spt Det. 15071
ATTN: DEH

HSC
Ft. Sam Houston AMC 78234
ATTN: HSLO-F
Fitzsimons AMC 80045
ATTN: HSHG-DEH
Walter Reed AMC 20307
ATTN: Facilities Engineer

INSCOM - Ch, Instl. Div.
Arlington Hall Station 22212
ATTN: Engr & Hsg Div
Vint Hill Farms Station 22186
ATTN: IAV-DEH
Ft. Belvoir 22060
ATTN: Facilities Engineer (3)

USA AMCCOM 61299
ATTN: AMSMC-RI
ATTN: AMSMC-IS

Military Dist of Washington
ATTN: DEH
Cameron Station (3) 22314
Fort Lesley J. McNair 20319
Fort Myer 22211

Military Traffic Mgmt Command
Falls Church 20315
Oakland Army Base 94626
Bayonne 07002
Sunny Point MOT 28461

NARADCOM, ATTN: DRDNA-F 01760

TARCOM, Fac, Div. 48090

TRADOC (19)
HQ, TRADOC, ATTN: ATEN-DEH 23651
ATTN: DEH

TSARCOM, ATTN: STSAS-F 63120

USAIS
Fort Huachuca 85613
ATTN: Facilities Engineer (3)
Fort Ritchie 21719

WESTCOM
Fort Shafter 96858
ATTN: DEH
ATTN: APEN-A

SHAPE 09055
ATTN: Survivability Sect. CCB-OPS
ATTN: Infrastructure Branch, LANDA

HQ USEUCOM 09128
ATTN: ECJ 4/7-LOE

Fort Belvoir, VA
ATTN: Australian Liaison Officer 22060
ATTN: Water Resource Center 22060
ATTN: Engr Studies Center 22060
ATTN: Engr Topographic Lab 22060
ATTN: ATZA-TE-SW 22060
ATTN: CECC-R 22060

CECRL, ATTN: Library 03755

CEWES, ATTN: Library 39180

HQ, XVIII Airborne Corps and
Ft. Bragg 28307
ATTN: AFZA-DEH-EE

Chanute AFB, IL 61868
3345 CES/DE, Stop 27

AMMRC 02172
ATTN: DRXMR-AF
ATTN: DRXMR-WE

Norton AFB, CA 92409
ATTN: AFRCE-MX/DE

Tyndall AFB, FL 32403
AFESC/Engineering & Service Lab

NAVFAC
ATTN: Division Offices (11)
ATTN: Facilities Engr Cmd (9)
ATTN: Naval Public Works Center (9)
ATTN: Naval Civil Engr Lab (3)
ATTN: Naval Constr Battalion Ctr 93043

Engineering Societies Library
New York, NY 10017

National Guard Bureau 20310
Installation Division

US Government Printing Office 20401
Receiving/Depository Section (2)

US Army Env. Hygiene Agency
ATTN: HSHB-ME 21010

Nat'l Institute of Standards & Tech 20899

Defense Technical Ino. Center 22304
ATTN: DTIC-FAB (2)

244
7/90

EMC Team Distribution

Chief of Engineers
ATTN: CEMP-ZA
ATTN: CEMP-ZM (2)
ATTN: DAEN-ZCP

CEHSC
ATTN: CEHSC-FB

US Army Engineer District
New York 10278
ATTN: Chief, Design Br
Pittsburgh 15222
ATTN: Chief, ORPCD
ATTN: Chief, Engr Div
Philadelphia 19106
ATTN: Chief, NAPEN-D
Baltimore 21203
ATTN: Chief, Engr Div
Norfolk 23510
ATTN: Chief, NAOEN-M
ATTN: Chief, NAOEN-D
Huntington 25701
ATTN: Chief, ORHED-G
Wilmington 28401
ATTN: Chief, SAWEN-D
Charleston 29402
ATTN: Chief, Engr Div
Savannah 31402
ATTN: Chief, SASAS-L
Jacksonville 32232
ATTN: Constr Branch
Mobile 36652
ATTN: Chief, SAMEN-D
ATTN: Chief, SAMEN-F
ATTN: Chief, SAMEN-C
Nashville 37202
ATTN: Chief, ORNED-P
Memphis 38103
ATTN: Chief, Constr Div
ATTN: Chief, LMMED-D
Vicksburg 39180
ATTN: Chief, Engr Div
Louisville 40201
ATTN: Chief, Engr Div
Detroit 48231
ATTN: Chief, NCEED-T
St. Paul 55101
ATTN: Chief, ED-D
ATTN: Chief, NCSED-GH
Rock Island 61204
ATTN: Chief, NCRED-G
St. Louis 63101
ATTN: Chief, ED-D
Kansas City 64106
ATTN: Chief, Engr Div
Omaha 68102
ATTN: Chief, Engr Div
New Orleans 70160
ATTN: Chief, LMNED-DG
Little Rock 72203
ATTN: Chief, Engr Div
Tulsa 74121
ATTN: Chief, Engr Div
Ft. Worth 76102
ATTN: Chief, SWFED-D
Galveston
ATTN: Chief, SWGAS-L 77553
ATTN: Chief, SWGCO-C 77553
ATTN: Chief, SWGED-DC 77550
Albuquerque 87103
ATTN: Chief, Engr Div
Los Angeles 90053
ATTN: Chief, SPLED-F
San Francisco 94105
ATTN: Chief, Engr Div
Sacramento 95814
ATTN: Chief, SPKED-D
ATTN: Chief, SPKCO-C
Far East 96301
ATTN: POFED-L
Portland 97208
ATTN: Geotech Engr Br
ATTN: Chief, FM-1
ATTN: Chief, EN-DB-ST
Seattle 98124
ATTN: Chief, NPSCO
ATTN: Chief, NPSEN-FM
ATTN: Chief, EN-DB-ST
Walla Walla 99362
ATTN: Chief, Engr Div
Alaska 99506
ATTN: NPAEN-G-M

US Army Engineer Division
New England 02154
ATTN: Chief, NEDED-T
ATTN: Laboratory
ATTN: Chief, NEDCD
North Atlantic 10007
ATTN: Chief, NADEN-T
Middle East (Rear) 22601
ATTN: Chief, MEDED-T
South Atlantic
ATTN: Laboratory 30060
ATTN: Chief, SADEN-TC 30303
ATTN: Chief, SADEN-TS 30303
Huntsville 35807
ATTN: Chief, HNDED-CS
ATTN: Chief, HNDED-M
ATTN: Chief, HNDED-SR
Lower Mississippi 39180
ATTN: Chief, LMVED-G
Ohio River
ATTN: Laboratory 45227
ATTN: Chief, Engr Div 45201
Missouri River
ATTN: Chief, MRDED-G 68101
ATTN: Laboratory 68102
Southwestern 75202
ATTN: Laboratory
ATTN: Chief, SWDED-MA
ATTN: Chief, SWDED-TG
South Pacific 94966
ATTN: Laboratory
Pacific Ocean 96858
ATTN: Chief, Engr Div
ATTN: FM&S Branch
ATTN: PODED-D
North Pacific
ATTN: Materials Lab 97060
ATTN: Chief, Engr Div 97208

7th US Army 09407
ATTN: AETTM-DTT-MG-EH

HQ, Combined Field Army (ROK/US)
ATTN: CFAR-EN 96358

US Army Foreign Sci. & Tech Ctr
ATTN: Charlottesville, VA 22901
ATTN: Far East Office 96328

USA ARRADCOM 07801
ATTN: DRDAR-LCA-OK

HQ, USAMRDC 21701
ATTN: SGRD-PLC

West Point, NY 10996
ATTN: Dept of Mechanics
ATTN: Library

Ft. Belvoir, VA 22060
ATTN: Learning Resources Center

Ft. Benning, GA 31905
ATTN: ATZB-DEH-BG
ATTN: ATZB-EH-E

Ft. Leavenworth, KS 66027
ATTN: ATZLCA-SA

Ft. Lee, VA 23801
ATTN: AMXMC-D (2)

Ft. McPherson, GA 30330
ATTN: AFEN-CD

Ft. Monroe, VA 23651
ATTN: ATEN-AD
ATTN: ATEN-FE-ME
ATTN: ATEN-FN (2)

Ft. Richardson, AK 99505
ATTN: AFVR-DE-E

Rocky Mountain Arsenal 80022
ATTN: SARRM-CO-FEP

USAWES 39180
ATTN: C/Structures
ATTN: Soils & Pavements Lab

Naval Facilities Engr Command 22332
ATTN: Code 2003

COMMANDER (CODE 2636) 93555
Naval Weapons Center

Little Rock AFB 72099
ATTN: 314/DEEE

Building Research Board 20418

Dept of Transportation Library 20590

Transportation Research Board 20418

108
+1
07/90